

**Amendments to the Specification**

Please amend paragraph [0211] of the publication as follows:

[0211] The parameters used with the above algorithms may be adapted depending on the sequence length and degree of homology studied. In some aspects, the parameters may be the default parameters used by the algorithms in the absence of instructions from the user. FIG. 2 is a flow diagram illustrating one aspect of a process 200 for comparing a new nucleotide or protein sequence with a database of sequences in order to determine the homology levels between the new sequence and the sequences in the database. The database of sequences can be a private database stored within the computer system 100, or a public database such as GENBANK that is available through the Internet. The process 200 begins at a start state 201 and then moves to a state 202 wherein the new sequence to be compared is stored to a memory in a computer system 100. As discussed above, the memory could be any type of memory, including RAM or an internal storage device. The process 200 then moves to a state 204 wherein a database of sequences is opened for analysis and comparison. The process 200 then moves to a state 206 wherein the first sequence stored in the database is read into a memory on the computer. A comparison is then performed at a state 210 to determine if the first sequence is the same as the second sequence. It is important to note that this step is not limited to performing an exact comparison between the new sequence and the first sequence in the database. Well-known methods are known to those of skill in the art for comparing two nucleotide or protein sequences, even if they are not identical. For example, gaps can be introduced into one sequence in order to raise the homology level between the two tested sequences. The parameters that control whether gaps or other features are introduced into a sequence during comparison are normally entered by the user of the computer system. Once a comparison of the two sequences has been performed at the state 210, a determination is made at a decision state 210 whether the two sequences are the same. Of course, the term "same" is not limited to sequences that are absolutely identical. Sequences that are within the homology parameters entered by the user will be marked as "same" in the process 200. If a determination is made that the two sequences are the same, the process 200

moves to a state 214 wherein the name of the sequence from the database is displayed to the user. This state notifies the user that the sequence with the displayed name fulfills the homology constraints that were entered. Once the name of the stored sequence is displayed to the user, the process 200 moves to a decision state 218 wherein a determination is made whether more sequences exist in the database. If no more sequences exist in the database, then the process 200 terminates at an end state 220. However, if more sequences do exist in the database, then the process 200 moves to a state 224 wherein a pointer is moved to the next sequence in the database so that it can be compared to the new sequence. In this manner, the new sequence is aligned and compared with every sequence in the database. It should be noted that if a determination had been made at the decision state 212 that the sequences were not homologous, then the process 200 would move immediately to the decision state 218 in order to determine if any other sequences were available in the database for comparison. Accordingly, one aspect of the invention is a computer system comprising a processor, a data storage device having stored thereon a nucleic acid sequence of the invention and a sequence comparer for conducting the comparison. The sequence comparer may indicate a homology level between the sequences compared or identify structural motifs, or it may identify structural motifs in sequences that are compared to these nucleic acid codes and polypeptide codes. FIG. 3 is a flow diagram illustrating one embodiment of a process 250 in a computer for determining whether two sequences are homologous. The process **250** begins at a start state **252** and then moves to a state **254** wherein a first sequence to be compared is stored to a memory. The second sequence to be compared is then stored to a memory at a state **256**. The process **250** then moves to a state **260** wherein the first character in the first sequence is read and then to a state **262** wherein the first character of the second sequence is read. It should be understood that if the sequence is a nucleotide sequence, then the character would normally be either A, T, C, G or U. If the sequence is a protein sequence, then it can be a single letter amino acid code so that the first and second sequences can be easily compared. A determination is then made at a decision state **264** whether the two characters are the same. If they are the same, then the process **250** moves to a state **268** wherein the next characters in the first and second sequences are read. A determination **270** is then made whether the next characters are the same. If

they are, then the process **250** continues this loop until two characters are not the same. If a determination **270** is made that the next two characters are not the same, the process **250** moves to a decision state **274** to determine whether there are any more characters either sequence to read. If there are not any more characters to read, then the process **250** moves to a state **276** wherein the level of homology between the first and second sequences is displayed to the user. If there are not any more characters to read, then the process **250** terminates **278**. The level of homology is determined by calculating the proportion of characters between the sequences that were the same out of the total number of sequences in the first sequence. Thus, if every character in a first 100 nucleotide sequence aligned with an every character in a second sequence, the homology level would be 100%.